AMENDMENTS TO THE SPECIFICATION

Please amend the indicated paragraphs of the Specification with amendments as indicated below:

[0005] Often, for example, when transferring confidential or financial data, it may be desirable for computing systems to communicate with one another in a secure manner. Accordingly, computing systems can authenticate and agree on the mechanisms used to secure the data (e.g., electronic messages) transferred between the computing systems. For example, two computing systems may use the Secure Sockets Layer ("SSL") protocol to establish a secure connection between one another. Further, it may be desirable for an application at a first computing system to prove proof of identity to another application at a second computing system (either in combination with or separately from computer system authentication). For example, a banking client may require that a corresponding ~~server~~ banking server prove its identity before the banking client will transfer financial data to the banking server. Proving the identity of an application can be done in accordance with Web service specifications, such as, for example, WS-Security and WS-Trust.

[0032] Generally, one module can provide another module with an indication of one or more measurable characteristics of the other module that are to be verified. For example, one module can request verification of measurable aspects of another module to cause the other module to prove it is appropriately configured to access a resource or to prove it is appropriately configured to issue challenges to other modules. It should be understood that any description of verifying measurable aspects to prove an appropriate configuration for accessing a resource also applies to verifying measurable aspects to prove an appropriate configuration for issuing challenges and vice-versa.

[0027] In some embodiments, proof of one or more measurable aspects of a requester are used along with machine authentication and application authentication to determine that a requester is appropriately configured to access a resource of a provider or that a provider is appropriately configured to issue challenges to a requester. For example, subsequent to two computing system authentication and/or two components of a distributed application authenticating, a requester may provide proof of one more aspects of the requester's configuration to a provider. Similarly, a provider may provide~~r~~ proof of one or more aspects of the provider's configuration to a requester. Solutions to challenges can be pre-computed and stored in a location accessible to a provider. Accordingly, when formulated proof is received from a requester, the formulated proof can be more efficiently verified.

[0074] The method 600 includes an act of accessing a first random value (act 601). For example, challenge service 510 can access seed nonce 501. The method 600 includes an act of accessing a secret value (act 602). For example, challenge service 510 can access secret 502. The method 600 includes an act of using the first random value and secret to generate a second random value (act 603). For example, hash algorithm 503 can ~~used~~ use seed nonce 501 and secret 502 to generate challenge nonce 504. Hash algorithm 503 ~~can~~ may use any of a wide variety of hash algorithms including the SHA1 algorithm.

[0075] The method 600 includes an act of using the first random value and the second random value to identify one or more regions within a portion of instructions (act 604). For example, hash algorithm 506 can use seed nonce 501 and challenge nonce 504 to identify regions 507 and 508. Hash algorithm 506 ~~can~~ may use any of a wide variety of hash algorithms including the P_SHA1 algorithm. Target 509 can include instructions for appropriately accessing resource 530. Target 509 can be, for example, an assembly, stored on computer-readable media, such as, for example, system memory or magnetic disk, that is accessible to challenge service 510.